VERIFYING PRIVACY-TYPE PROPERTIES IN A MODULAR WAY

M. Arapinis (1), V. Cheval (2), S. Delaune (2)
(1) School of Computer Science, Birmingham, UK
(2) LSV, ENS Cachan, CNRS, INRIA Saclay

ANR ProSe

30 October 2012
To verify security properties on protocols, we model protocols in isolation.

Possible problems:
- Protocols may share same keys
- Protocols may share same cryptographic primitives
- Tools may not be able to prove the security property
Our goal

Verifying $S$ on $P$ and

Verifying $S$ on $Q$

where

- $P$ and $Q$ may share secrets and cryptographic primitives
- $S$ is a security property
| Security properties |  |
### Context

**Security properties**

**Reachability properties**
- Secrecy, Authentication, ...
Security properties

- **Reachability properties**
  - Secrecy, Authentication, ...

- **Equivalence properties**
  - Anonymity, Privacy, Receipt-Freeness, ...
Example of equivalence property: anonymity

CONTEXT

Alice

Intruder

Unknown
Example of equivalence property: anonymity

Can the intruder distinguish the two situations?
PREVIOUS WORKS

- On reachability properties

- On equivalence properties: Tagged protocol
MOTIVATION

Privacy-type properties: Anonymity and unlinkability

Concrete example: e-passport protocols
- Basic Access Control (BAC): establishes sessions keys between reader and a passport
- Passive Authentication (PA)
- Active Authentication (AA)

Passive Authentication and Active Authentication are executed in parallel
Composition context for anonymity

\[ P : A \rightarrow S : \{id_A\}_r^{\text{pk}}(k_S) \]
FORMALISM

Composition context for anonymity

\[
P : A \rightarrow S : \{id_A\}_{pk(k_S)}^r
\]

Definition from: M. Arapinis, T. Chothia and M. Ryan. *Analysing unlinkability and anonymity using the applied pi calculus.*
Composition context for anonymity

\[
P : A \to S : \{id_A\}^r_{pk(k_S)}
\]

\[
C'[\_] \triangleq \text{new } k_S. !\text{new } id_A. !\_
\]

\[
C[_1, _2] \triangleq \text{new } k_S. ((!\text{new } id_A. !_1) | !_2)
\]

\[
C[P, P\{i^d_A / i^d_A\}] \approx C'[P]
\]

Definition from: M. Arapinis, T. Chothia and M. Ryan. *Analysingunlinkability and anonymity using the applied pi calculus.*
**FORMALISM**

Composition context for anonymity

\[
P : \ A \rightarrow S : \ \{id_A\}_r^{pk(k_S)}
\]

\[
C'[: ] \overset{\text{def}}{=} \text{new } k_S . !\text{new } id_A . !\_
\]

\[
C[: -1 , -2 ] \overset{\text{def}}{=} \text{new } k_S . ( (!\text{new } id_A . !_-1 ) | !-_2 )
\]

\[
C[P , P \{id_o / id_A \} ] \approx C' [P]
\]

\[
C[Q , Q \{id_o / id_A \} ] \approx C' [Q]
\]

Definition from: M. Arapinis, T. Chothia and M. Ryan. *Analysing unlinkability and anonymity using the applied pi calculus.*
FORMALISM

Composition context for anonymity

\[ P : A \rightarrow S : \{id_A\}^{r}_{pk(k_S)} \]

\[ C'[\_] \overset{\text{def}}{=} \text{new } k_S. \! \text{new } id_A. \! \_ \]

\[ C[\_1, \_2] \overset{\text{def}}{=} \text{new } k_S.((\! \text{new } id_A. \! \_1) | \! \_2) \]

\[ C[P, P\{id_O / id_A\}] \simeq C'[P] \]

\[ C[Q, Q\{id_O / id_A\}] \simeq C'[Q] \]

\[ C[Q | P, (Q | P)\{id_O / id_A\}] \simeq C'[Q | P] \]

Definition from: M. Arapinis, T. Chothia and M. Ryan. *Analysing unlinkability and anonymity using the applied pi calculus.*
CONDITIONS

No shared key revealed

\[ P : A \rightarrow S : \{id_A\}_r^{pk(k_S)} \]
\[ Q : S \rightarrow A : k_S \]

\( P \) preserves the anonymity of A
\( Q \) preserves the anonymity of A

\( P \mid Q \) does not preserve the anonymity of A
 Tag shared cryptographic primitives

\[
P : \quad A \rightarrow S : \{id_A\}_r^{pk(k_S)}
\]
\[
Q : \quad A \rightarrow S : \{Na\}_r^{pk(k_S)}
\]
\[
S \rightarrow A : Na
\]

\[P\] preserves the anonymity of A
\[Q\] preserves the anonymity of A

\[P \mid Q\] does not preserve the anonymity of A
Public key revealed at the beginning

\[ P_i : \ A \rightarrow S : \ \{ \text{tag}_a(id_i) \}_{\text{pk}(k_S)} \]

\[ Q : \ S \rightarrow A : \text{pk}(k_S) \]

\[ C[\_] \overset{\text{def}}{=} \text{new } k_S. \_ \]
**CONDITIONS**

Public key revealed at the beginning

\[ P_i : A \rightarrow S : \{ \text{tag}_a(id_i) \} \text{pk}(k_S) \]

\[ Q : S \rightarrow A : \text{pk}(k_S) \]

\[ C[\_] \overset{\text{def}}{=} \text{new } k_S. \_ \]

\[ C[P_1] \approx C[P_2] \quad \text{and} \quad C[Q] \approx C[Q] \]

But \[ C[P_1 | Q] \not\approx C[P_2 | Q] \]
If:
- The shared keys of C and C' are not revealed
- The public keys are revealed at the beginning
- The protocols A and B are tagged
Passive Authentication (PA)

Passport Tag

\[ \text{ksenc, ksmac, } sk_P \]

Reader

\[ \text{ksenc, ksmac, } \text{vk}(sk_P) \]

\[ xenc \leftarrow \text{senc}(\text{read, ksec}) \]
\[ xmac \leftarrow \text{mac}(xenc, ksmac) \]

\[ \langle xenc, xmac \rangle \]

\[ yenc \leftarrow \text{senc}(\langle dg_1, \ldots, dg_{19}, sod \rangle, ksec) \]
\[ ymac \leftarrow \text{mac}(yenc, ksmac) \]

\[ \langle yenc, ymac \rangle \]
E-PASSPORT

Active Authentication (AA)

Passport Tag
\(k_{senc}, k_{smac}, sk_P\)

Reader
\(k_{senc}, k_{smac}, \text{vk}(sk_P)\)

\[\text{new } \text{rand}\]
\[x_{enc} \leftarrow \text{senc}(\langle \text{init}, \text{rand} \rangle, k_{senc})\]
\[x_{mac} \leftarrow \text{mac}(x_{enc}, k_{smac})\]

\(\langle x_{enc}, x_{mac}\rangle\)

\[\text{new } \text{nce}\]
\[\sigma \leftarrow \text{sign}(\langle \text{nce}, \text{rand} \rangle, sk_P)\]
\[y_{enc} \leftarrow \text{senc}(\sigma, k_{senc})\]
\[y_{mac} \leftarrow \text{mac}(y_{enc}, k_{smac})\]

\(\langle y_{enc}, y_{mac}\rangle\)
With ProVerif,
- we prove anonymity for $AA$
- we can not prove anonymity for $PA$
- we can not prove anonymity for $PA \mid AA$
E-PASSPORT

With ProVerif,

• we prove anonymity for $AA$
• we can not prove anonymity for $PA$
• we can not prove anonymity for $PA \mid AA$

proving anonymity for $PA$

implies

proving anonymity for $PA \mid AA$
$C[P_A] \approx C'[P'_A]$ and $C[P_B] \approx C'[P'_B]$
SKETCH OF PROOF

\[ C[P_A] \approx C'[P'_{A}] \quad \text{and} \quad C[P_B] \approx C'[P'_{B}] \]

\[ C[P_A] \mid C[P_B] \approx C'[P'_{A}] \mid C'[P'_{B}] \]
SKETCH OF PROOF

\[ C[P_A] \approx C'[P'_A] \quad \text{and} \quad C[P_B] \approx C'[P'_B] \]

\[ C[P_A] \mid C[P_B] \approx C'[P'_A] \mid C'[P'_B] \]

\[ C[P_A \mid P_B] \approx C'[P'_A \mid P'_B] \]
SKETCH OF PROOF

\[ C[P_A] \approx C'[P'_A] \quad \text{and} \quad C[P_B] \approx C'[P'_B] \]

\[ C[P_A] \mid C[P_B] \approx C'[P'_A] \mid C'[P'_B] \]

\[ \approx \]

\[ C[P_A \mid P_B] \]
$C[P_A] \approx C'[P'_A]$ and $C[P_B] \approx C'[P'_B]$

$C[P_A] \ | \ C[P_B] \approx C'[P'_A] \ | \ C'[P'_B]$

$\approx$

$C'[P'_A \ | \ P'_B]$
\[ \begin{align*}
C[P_A] & \approx C'[P'_A] \quad \text{and} \quad C[P_B] \approx C'[P'_B] \\
C[P_A] \mid C[P_B] & \approx C'[P'_A] \mid C'[P'_B] \\
\approx & \\
C[P_A \mid P_B] & \approx C'[P'_A \mid P'_B]
\end{align*} \]
\[ C[P_A] \mid C[P_B] \cong C[P_A \mid P_B] \]

\[ \text{new} \ k.[P_A \mid P_B] \]

\[ \text{new} \ k.P_A \mid \text{new} \ k.P_B \]
SKETCH OF PROOF

\[ C[PA] \mid C[PB] \cong C[PA \mid PB] \]

\[
\text{new } k.[PA \mid PB] \quad \rightarrow \quad P_1 \quad \rightarrow \quad P_n
\]

\[
\text{new } k.P_A \mid \text{new } k.P_B
\]
$C[PA] \mid C[PB] \approx C[PA \mid PB]$

c new $k. [PA \mid PB] \quad \rightarrow \quad P_1 \quad \rightarrow \quad P_n$

new $k.P_A \mid$ new $k.P_B$
$C[P_A] \mid C[P_B] \approx C[P_A \mid P_B]$

new $k. [P_A \mid P_B] \rightarrow P_1 \rightarrow \cdots \rightarrow P_n$

$C[P_A] \mid C[P_B] \approx C[P_A \mid P_B]$

new $k. P_A \mid new k. P_B$
\[
C[P_A] \mid C[P_B] \simeq C[P_A \mid P_B]
\]

new \( k.[P_A \mid P_B] \) \( \rightarrow \) \( P_1 \rightarrow \cdots \rightarrow P_n \)

\[
C\left[\text{new } k.P_A \mid \text{new } k.P_B\right] \rightarrow \delta(P_1) \rightarrow \delta(P_n)
\]
CONCLUSION & FUTURE WORK

- Parallel composition theorem for equivalence properties

  Conditions:
  - The shared keys are not revealed
  - The public keys are revealed at the beginning
  - The protocols are tagged

- Future work: Sequential composition

  E-passport protocols
  - Basic Access Control (BAC): establishes sessions keys between reader and a passport
    - Passive Authentication (PA)
    - Active Authentication (AA)

- Future work: Removing the tags

  - Tags imply heavy transformation of the protocol
  - Almost no current protocol tags all their message
  - Protocols may behave as if they were tagged (ex: nonce exchange)